

02/01/00
JC772 U.S.P.T.O.
PTO/SB/05 (Rev. 08/01/99)

A
U.S.P.T.O.
PTO/SB/05 (Rev. 08/01/99)

Please type a plus sign (+) inside this box →

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

UTILITY PATENT APPLICATION TRANSMITTAL

(Only for new nonprovisional applications under 37 C.F.R. § 1.53(b))

Attorney Docket No.	6847 US 1
First Inventor or Application Identifier	THUMPUDI NAVEEN
Title	Coarse Representation of Visual Objects...
Express Mail Label No.	EL415462566US

APPLICATION ELEMENTS

See MPEP chapter 600 concerning utility patent application contents.

1. * Fee Transmittal Form (e.g., PTO/SB/17)
(Submit an original and a duplicate for fee processing)
2. Specification [Total Pages]
 - Descriptive title of the Invention
 - Cross References to Related Applications
 - Statement Regarding Fed sponsored R & D
 - Reference to Microfiche Appendix
 - Background of the Invention
 - Brief Summary of the Invention
 - Brief Description of the Drawings (if filed)
 - Detailed Description
 - Claim(s)
 - Abstract of the Disclosure
3. Drawing(s) (35 U.S.C. 113) [Total Sheets]
4. Oath or Declaration [Total Pages]
 - a. Newly executed (original or copy)
 - b. Copy from a prior application (37 C.F.R. § 1.63(d))
(for continuation/divisional with Box 16 completed)
 - i. DELETION OF INVENTOR(S)
Signed statement attached deleting inventor(s) named in the prior application, see 37 C.F.R. §§ 1.63(d)(2) and 1.33(b).

NOTE FOR ITEMS 1, 8 & 13: IN ORDER TO BE ENTITLED TO PAY SMALL ENTITY FEES, A SMALL ENTITY STATEMENT IS REQUIRED (37 C.F.R. § 1.27). EXCEPT IF ONE FILED IN A PRIOR APPLICATION IS RELIED UPON (37 C.F.R. § 1.28).

16. If a CONTINUING APPLICATION, check appropriate box, and supply the requisite information below and in a preliminary amendment:
 Continuation Divisional Continuation-in-part (CIP)
of prior application No.: 60 / 118,386

Prior application information: Examiner _____

Group / Art Unit: _____

For CONTINUATION or DIVISIONAL APPS only: The entire disclosure of the prior application, from which an oath or declaration is supplied under Box 4b, is considered a part of the disclosure of the accompanying continuation or divisional application and is hereby incorporated by reference. The incorporation can only be relied upon when a portion has been inadvertently omitted from the submitted application parts.

17. CORRESPONDENCE ADDRESS

<input type="checkbox"/> Customer Number or Bar Code Label	(Insert Customer No. or Attach bar code label here)				<input checked="" type="checkbox"/> Correspondence address below
Name	Peter D. Symes				
Address	Grass Valley (U.S.) Inc. P.O. Box 599000 400 Providence Mine Road				
City	Nevada City	State	California	Zip Code	95959-7900
Country	U.S.A.	Telephone	530 478-3437	Fax	

Name (Print/Type)	Francis I. Gray	Registration No. (Attorney/Agent)	27,788
Signature	Francis I. Gray	Date	Feb. 1, 2000

Burden Hour Statement: This form is estimated to take 0.2 hours to complete. Time will vary depending upon the needs of the individual case. Any comments on the amount of time you are required to complete this form should be sent to the Chief Information Officer, Patent and Trademark Office, Washington, DC 20231. DO NOT SEND FEES OR COMPLETED FORMS TO THIS ADDRESS. SEND TO: Assistant Commissioner for Patents, Box Patent Application, Washington, DC 20231.

TITLE OF THE INVENTION

**COARSE REPRESENTATION OF VISUAL OBJECT'S SHAPE
FOR SEARCH/QUERY/FILTERING APPLICATIONS**

5

CROSS REFERENCE TO RELATED APPLICATIONS

This is a continuation of provisional U.S. Patent Application Serial No. 60/118,386 filed February 1, 1999, now abandoned.

10 BACKGROUND OF THE INVENTION

The present invention relates to video data processing, and more particularly for a coarse representation of a visual object's shape for search/query/filtering applications.

With the success of the Internet and picture and video coding standards, such as JPEG, MPEG-1, 2, more and more audio-visual information is available in digital form. Before one can use any such information, however, it first has to be located. Searching for textual information is an established technology. Many text-based search engines are available on the World Wide Web to search text documents. Searching is not yet possible for audio-visual content, since no generally recognized description of this material exists. MPEG-7 is intending to standardize the description of such content. This description is intended to be useful in performing search at a very high level or at a low level. At a high level the search may be to locate "a person wearing a white shirt walking behind a

person wearing a red sweater". At lower levels for still images one may use characteristics like color, texture and information about the shape of objects in that picture. The high level queries may be mapped to the low level primitive queries to perform the search.

5 Visual object searches are useful in content creation, such as to locate from archive the footage from a particular event, e.g. a tanker on fire, clips containing particular public figure, etc. Also the number of digital broadcast channels is increasing every day. One search/filtering application is to be able to select the broadcast channel (radio or TV) that is potentially interesting.

10 What is desired is a descriptor that may be automatically or semi-automatically extracted from still images/key images of video and used in searches.

15 BRIEF SUMMARY OF THE INVENTION

Accordingly the present invention provides a coarse representation of a visual object's shape for search/query/filtering applications.

The objects, advantages and other novel features of the present invention are apparent from the following detailed description in view of the appended claims and attached drawing.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWING

Fig. 1 is an illustrative view of a visual object within a digital image.

Fig. 2 is an illustrative view of the elements of a feature vector according to the present invention.

Fig. 3 is a block diagram view of a feature vector extraction process according to the present invention.

Fig. 4 is a block diagram view of a search engine based upon coarse shape feature vectors.

DETAILED DESCRIPTION OF THE INVENTION

A coarse representation of a visual object's shape may be used for searching based on the shape of the object. This representation is easy to compute, but answers a variety of queries that will be described later.

However, this simple approach may not provide a very high quality shape representation, either in 2-D or 3-D. The following method may be used for visual objects in still images or in video.

As shown in Fig. 1 in a coarse representation of shape, each semantic object or its sub-portions may be represented by a binding box (a

rectangle) in the image. A binding box of a visual object is the tightest rectangle that fully encompasses that visual object in an image.

The parameters needed to represent the binding box are:

5	TopLeftCorner _h
10	TopLeftCorner _v
	BoxWidth
	BoxHeight
	FractionalOccupancy

As shown in Fig. 2 the position components TopLeftCorner_h and TopLeftCorner_v of the binding box are defined as offsets in horizontal and vertical directions with respect to the origin of the picture, which is nominally at the top-left corner of the image. The FractionalOccupancy is a number between 0 and 1. This is the fraction of samples in the binding box that belong to the object being described. In order to describe TopLeftCorner_h, TopLeftCorner_v, BoxWidth and BoxHeight, a normalized coordinate system is used. In this system the height of the image when displayed is normalized to 1.0. Subsequently, the display width is measured in units of display height. As an example, for a 320x240 image that uses square pixels for display, the height of 240 pixels is mapped to 1.0, and the width of 320 is mapped to 1.333 (320/240).

25	0.43
	0.51
An example feature vector is	0.22
	0.25
30	0.83

Low level queries served by this feature vector include:

1. Find the visual objects that have a particular aspect ratio (ratio of height to width).
2. Find the visual objects that are at least x% (a given percentage) of the picture size.
3. Find the visual objects that are at most x% (a given percentage) of the picture size.
4. Find the visual objects that are positioned near (x,y) (a particular coordinate) location in the picture.
5. Find the visual objects that are at least x% (a given percentage) dense.
6. Find the visual objects that are at most x% (a given percentage) dense.
7. Find the visual objects that have at least "y" units height.
8. Find the visual objects that have at most "y" units height.
9. Find the visual objects that have at least "x" units width.
10. Find the visual objects that have at most "x" units width.
11. Estimating the trajectory of a particular visual object in time, in a given video.

Overall extraction of a coarse representation of a visual object's shape is shown in Figure 3. The steps involved are (1) segmentation, (2) extraction of the bitmap of object of interest, and finally (3) estimation of the binding box. In this figure, the segmentation process may either be

5 automatic, semi-automatic, or. The segmentation map consists of segmentation labels at each pixel. The set of pixels having a particular segmentation label belong to a distinct visual object. Thus, the second stage merely creates a binary map, with values "valid" (true, 1, or 255) wherever segmentation label equals an objectID of interest, and "invalid" 10 (false, or 0) elsewhere. Identification of the largest connected region in the bitmap is covered in co-pending provisional U.S. Patent Application Serial No. 60/118, The binding box estimation procedure gets as input the bitmap indicating the validity of each pixel and the display aspect ratio that is right for the picture.

15 The process of estimating the binding box itself may be broken down as:

1. Estimating in pixel units the TopLeftCorner_h , TopLeftCorner_v , BoxWidth , BoxHeight , and $\text{FractionalOccupancy}$.
2. Normalizing the units.

20 The estimation of TopLeftCorner_h , TopLeftCorner_v , BoxWidth , BoxHeight , and $\text{FractionalOccupancy}$ is performed by the following C++

code segment. The inputBitmap is a 2-D array that contains the validity of each sample (i.e. does it belong to the object of interest or not) information.

```
int botRightv, botRighth;  
5      int i, j, nr, nc, nSamples = 0;  
  
occupancy = 0;  
  
boxWidth = boxHeight = topLeftv = 0;  
  
topLefth = botRightv = botRighth = 0;  
  
bool valid;  
10     nr = imageHeight;  
  
nc = imageWidth;  
  
// topLeftv  
  
valid = false;  
  
for (i = 0 ; i < nr; i++) {  
  
    for (j = 0; j < nc; j++) {
```

-8-

```
if (inputBitmap[i][j] is valid) {  
    valid = true;  
  
    break;  
}  
  
5    }  
  
if (valid) break;  
  
}  
  
topLeftv = i;  
  
// topLefth  
  
10   valid = false;  
  
for (j = 0; j < nc; j++) {  
  
    for (i = 0 ; i < nr; i++) {  
  
        if (inputBitmap[i][j] is valid) {  
            valid = true;  
  
            break;  
}  
}
```

```
    }

    if (valid) break;

}

topLefty = j;

5      // botRightv

    valid = false;

    for (i = nr-1 ; i >= 0; i--) {

        for (j = 0; j < nc; j++) {

            if (inputBitmap[i][j] is valid) {

                10      valid = true;

                break;
            }
        }

        if (valid) break;
    }

    15      }

    botRightv = i;
```

```
// botRighth  
  
    valid = false;  
  
    for (j = nc-1; j >= 0; j--) {  
  
        for (i = 0 ; i < nr; i++) {  
  
            if (inputBitmap[i][j] is valid) {  
  
                valid = true;  
  
                break;  
  
            }  
  
        }  
  
        if (valid) break;  
  
    }  
  
    botRighth = j;  
  
    for (i = topLeftv; i <= botRightv; i++)  
  
        for (j = topLefth; j <= botRighth; j++)  
  
            if (inputBitmap[i][j] is valid) nSamples++;  
  
    if (nSamples > 0) {
```

```
    boxHeight = botRightv - topLeftv + 1;  
  
    boxWidth = botRighth - topLefth + 1;  
  
    occupancy = double(nSamples) /double(boxHeight *  
    boxWidth);  
  
5      }  
  
10
```

Display aspect ratio (DAR) is the ratio of the height of the displayed picture to the width of the displayed picture, say in meters. For example, it is 3/4 for conventional TV, 9/16 for HDTV. Given the estimated results (from above) that are in pixel units, the following relations may be used to perform normalization of the units.

$$\text{NormBoxHeight} = \text{PixelBoxHeight} / \text{Pixel Picture Height}$$

$$\text{NormBoxWidth} = \text{PixelBoxWidth} / (\text{PixelPictureWidth} * \text{DAR})$$

$$\text{NormTopLeftCorner}_v = \text{PixelTopLeftCorner}_v / \text{PixelPictureHeight}$$

$$\text{NormTopLeftCorner}_h = \text{PixelTopLeftCorner}_h / (\text{PixelPictureHeight} * \text{DAR})$$

$$15 \quad \text{NormFractionalOccupancy} = \text{PixelFractionalOccupancy}$$

Once the feature vectors are available for each visual object in each image of the database, it is quite trivial to perform a matching/query process based on the queries listed above. A search engine shown in Fig. 4 compares the query number to the appropriate element of the feature

vectors, and performs sorting to pick the best matches. In these searches, the search engine needs additional metadata: the display aspect ratio, width and height in pixels.

Here details are provided for the particular query " Find the visual
5 objects that have an aspect ratio (ratio of height to width) of A". In
response to this query, the search engine:

1. Computes the aspect ratios (α_i) of all the visual objects in the database (i.e. ratio of BoxHeight to BoxWidth),
2. Computes the Euclidean distance d_i from α_i to A for each i. Other
10 distance metrics are also possible.

$$d_i = |A - \alpha_i|$$

3. Sorts d_i in descending order.
4. Presents the top results in the sorting to the user who made the
query.

15 The search engine can pre-compute a lot of information to speed-up
the search.

Thus the present invention provides a coarse representation of a visual object's shape for search/query/filtering applications by representing each object by a binding box.

CLAIM OR CLAIMS

WHAT IS CLAIMED IS:

1. A method of course representation of the shape of a visible object in a

5 digital picture comprising the steps of:

segmenting visible objects from the digital picture;

extracting a bitmap for an object of interest from the segmented
visible objects; and

estimating from the bitmap and a display aspect ratio a binding box
10 for the object of interest.

2. The method as recited in claim 1 wherein the estimating step comprises
the steps of:

estimating in pixel units a set of parameters for the binding box; and

15 normalizing the pixel units to form a feature vector representing the
binding box.

3. The method as recited in claim 2 further comprising the step of
searching a video database having visible objects, each visible object

20 having an associated feature vector, to find those visible objects whose
feature vectors match the feature vector of the object of interest.

4. The method as recited in claim 3 wherein the searching step comprises

the steps of:

computing aspect ratios for all visible objects in the video database;

computing distances according to a specified distance metric

between the desired aspect ratio and the aspect ratios for the visible

5 objects in the video database;

sorting the distances in descending order to produce a sort list of aspect ratios and associated visible objects; and

displaying the visible objects associated with the aspect ratios that are at the top of the sort list.

ABSTRACT OF THE DISCLOSURE

A method of coarse representation of a visual object's shape for search/query/filtering applications uses a binding box that fully encompasses the object of interest within the image to extract a feature vector. Once the feature vector is available, matching based on specific queries may be performed using a search engine to compare the query number to an appropriate element of the feature vector, performing sorting to pick the best matches.

10

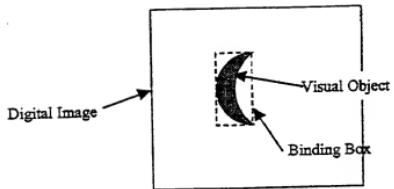


Figure 1

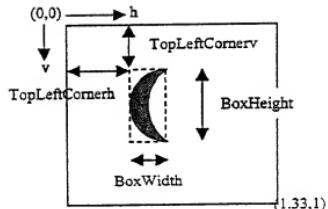


Figure 2

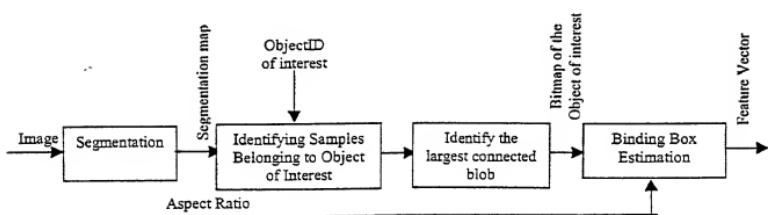


Figure 3

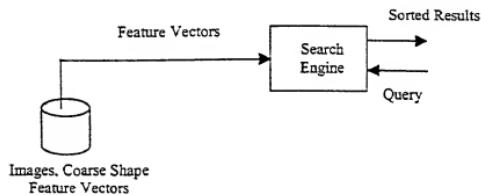


Figure 4